REPORT DOCUMENTATION PAGE

AFRL-SR-BL-TR-00-

0679

1. REPORT DATE *(DD-MM-YYYY)* 30-08-2000	**2. REPORT TYPE** Final technical report	**3. DATES COVERED** *(From - To)* 1 April 1997 – 31 May 2000

4. TITLE AND SUBTITLE
Constructing and Refining Knowledge Bases:
A Collaborative Apprenticeship Multistrategy Learning Approach

5a. CONTRACT NUMBER

5b. GRANT NUMBER
F49620-97-1-0188

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
Gheorghe Tecuci

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Learning Agents Laboratory, MSN 4A5
Department of Computer Science
School of Information Technology and Engineering
George Mason University
4400 University Dr.
Fairfax, VA 22030-4444

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Air Force Office of Scientific Research
801 N. Randolph St. Room 732
Arlington VA 22203-1977

10. SPONSOR/MONITOR'S ACRONYM(S)
AFOSR/NM

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION / AVAILABILITY STATEMENT

Approved for public release,
distribution unlimited

20010109 045

13. SUPPLEMENTARY NOTES

14. ABSTRACT

This research has developed a theory, methodology and learning agent shell for development of knowledge bases and knowledge-based agents, by domain experts, with limited assistance from knowledge engineers. The main feature of this approach is that a subject matter expert communicates his or her expertise to the learning agent in a very natural way, similar to how the expert would communicate it to a human apprentice while solving problems in cooperation. Starting from an initial ontology, an expert may teach the agent how to solve a certain type of problem by providing a concrete example, helping the agent to understand the solution, supervising the agent as it attempts to solve new problems, and correcting its mistakes. Through such natural interactions, the agent will be guided in learning complex problem solving rules, and in extending and correcting its knowledge base. This research has been done in the context of the DARPA High Performance Knowledge Bases program, where it has been applied to two challenge problems, the Workaround challenge problem, and the Course of Action challenge problem. The agent development approach and the two developed agents have been evaluated in several intensive studies, and have demonstrated very good results.

15. SUBJECT TERMS
artificial intelligence, knowledge acquisition, multistrategy machine learning, knowledge engineering, knowledge base, knowledge-based agent

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON**
					Gheorghe Tecuci
a. REPORT non classified	**b. ABSTRACT** non classified	**c. THIS PAGE** non classified	none	16	**19b. TELEPHONE NUMBER** *(include area code)* 703 993 1722

1. Introduction

This research project has addressed the problem of acquiring knowledge from a subject matter expert and representing it in the knowledge base of an intelligent agent, with limited assistance from a knowledge engineer. The research has been done in the context of the DARPA High Performance Knowledge Bases (HPKB) Program, where it has been applied to two challenge problems, the Workaround challenge problem and the Course of Action challenge problem.

The next section summarizes the main results of this research. After the title of each contribution there are numeric references to the published papers listed in section 3.

Section 4 lists the presentations and the demonstrations of the performed research that have been made at the AFOSR PI meetings and at the DARPA meetings organized as part of the HPKB program. This list also includes several invited talks. However, it does not include the presentations and the demonstrations of the conference papers listed in section 3.

Section 5 lists the most significant events, achievements and interactions with other organizations, which took place during this research project.

Finally, section 6 lists the personnel associated with this research.

2. Summary of the main contributions

The Disciple approach ([13])

This research has developed a theory, methodology and system, called Disciple, that allows a subject matter expert that has little knowledge engineering or computer experience, to build a knowledge base and a knowledge-based agent, with limited assistance from a knowledge engineer. Starting from an initial ontology, an expert may teach the agent how to perform various tasks, in a way that resembles how the expert would teach a human apprentice when solving problems in cooperation. During this process, the agent learns from the expert, building, verifying and improving its knowledge base.

The general strategy behind the Disciple approach is to replace the difficult knowledge engineering tasks required to build a knowledge base, which cannot be performed by a subject matter expert, with simpler tasks that can be performed by the expert, as shown in Figure 1.

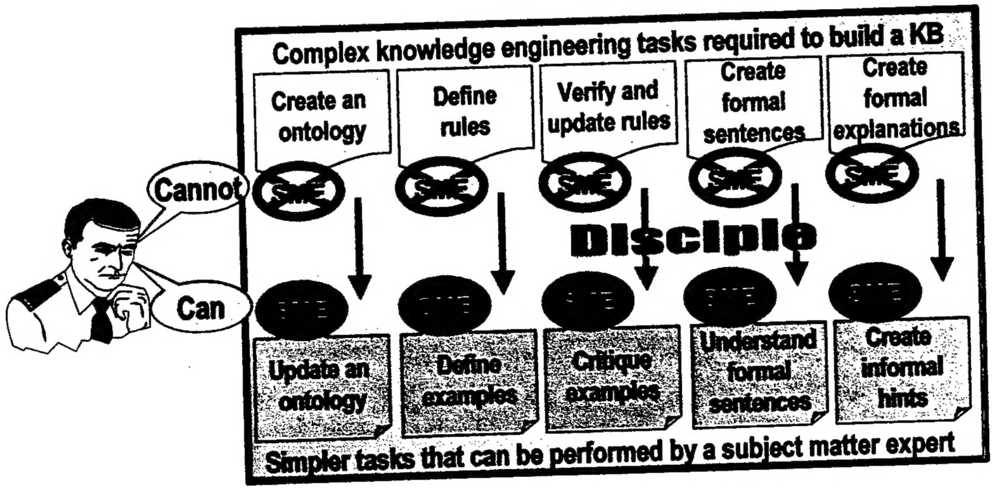

Figure 1: The general strategy behind the Disciple approach

To build a knowledge base one has to first create an ontology that defines the terms of the application domain. Then one has to define problem solving rules or methods, verify and update them. In general, these tasks require the creation of formal sentences and formal explanations.

In the Disciple approach, these tasks are replaced with simpler tasks that can be performed by a subject matter expert, with limited support from a knowledge engineer. Instead of creating an ontology, the expert need only update and extend an initial ontology imported from existing repositories of knowledge. Instead of defining a complex problem solving rule, the expert need only define a specific example of a problem solving episode because Disciple will be able to learn a rule from that example. Instead of debugging a complex problem solving rule, the expert need only critique specific examples of problem solving episodes and Disciple will update the corresponding rule accordingly. Most of the time, the expert will not need to create formal sentences, just understand such sentences generated by Disciple. Also, the expert will not need to provide formal explanations, just informal hints that will guide Disciple to generate plausible explanations from which the expert will choose the correct ones.

The Disciple approach is based on several levels of synergism between the expert that has the knowledge to be formalized and the agent that is able to formalize it. At the highest level there is the synergism in solving complex problems, where the agent contributes routine and innovative problem solving steps and the expert contributes creative ones. At the next level down, there is the synergism between teaching and learning, where the expert helps the agent understand the problem solving steps contributed by him or her, and the agent learns general problem solving rules that will allow it to apply similar steps in future problem solving situations. Finally, at the lowest level, there is the synergism between different learning strategies employed by the agent to learn from the expert in situations in which no single strategy learning method would be sufficient.

These levels of synergism have been made possible by the employment of an original representation of problem solving knowledge, called a plausible version space rule. Such

a rule includes both a plausible lower bound and a plausible upper bound for the problem solving knowledge, allowing a natural integration of problem solving, learning and teaching.

A comprehensive description of the Disciple approach is the subject of the book:
Tecuci G., *BUILDING INTELLIGENT AGENTS: An Apprenticeship Multistrategy Learning Theory, Methodology, Tool and Case Studies,* Academic Press, 1998.

This book is the best reference for the basic Disciple theory and methodology, knowledge representation, elementary problem solving methods, rule learning method, rule refinement method, and exception-handling methods. Each of these aspects, however, have been further developed and have been presented in more recent publications, as will be summarized below.

A methodology for developing intelligent agents ([20], [23])

The developed Disciple methodology for building end-to-end agents consists of the following steps: 1) Specification of the problem 2) Modeling of the problem solving process as task reduction. 3) Customization of the general Disciple shell for the specific application domain. 4) Importing of ontological knowledge from existing repositories. 5) Extending and updating of the ontology. 6) Training of the agent for its domain-specific tasks. 7) Testing and using of the agent.

The Disciple methodology and shell provide solutions to some of the issues that have been found to be limiting factors in developing knowledge-based agents: limited ability to reuse previously developed knowledge; the knowledge acquisition bottleneck; the knowledge adaptation bottleneck; the scalability of the agent building process; finding the right balance between using general tools and developing domain specific modules; and the portability of the agent building tools and of the developed agents.

This methodology was successfully applied for the development of two agents for solving two of the HPKB challenge problems, as summarized below.

An architecture for a learning agent shell ([1], [20])

A result of this research is the development of the concept of a *learning agent shell* as a new class of tools for rapid development of practical end-to-end knowledge-based agents, by domain experts, with limited assistance from knowledge engineers. A learning agent shell consists of a learning and knowledge acquisition engine as well as an inference engine and supports building an agent with a knowledge base consisting of an ontology and a set of problem solving rules.

Disciple is an example of such a learning agent shell. Disciple-workaround and Disciple-COA, presented in the following sections, are customizations of the general Disciple shell.

A Disciple-based solution to the workaround challenge problem ([23])

The Disciple approach has been developed and scaled-up for application to the workaround challenge problem defined in the DARPA High Performance Knowledge Bases program. The workaround challenge problem consists of estimating the enemy's best way of working around damage to the transportation infrastructure, such as a damaged bridge or a cratered road. By solving the workaround challenge problem it has been demonstrated that a knowledge engineer can use Disciple to rapidly build and update a knowledge base by capturing knowledge from military engineering manuals and a set of sample solutions provided by a subject matter expert.

During the July 1998 annual evaluation of the workaround knowledge bases developed in the HPKB project, the Disciple agent demonstrated the highest rate of knowledge acquisition and the best quality of the generated solutions. These results are reported in the December-98 issue of AI Magazine (Cohen P., Schrag R., Jones E., Pease A., Lin A., Starr B., Gunning D., and Burke M. 1998. The DARPA High-Performance Knowledge Bases Project, *AI Magazine*, 19(4),25-49).

Based on its solution to the workaround challenge problem, Disciple was selected to be included in an experiment at EFX'98, the Air Force's annual showcase for promising new technologies, that took place in September 1998. For this experiment, the Disciple workaround generator was further extended and delivered to Alphatech. Alphatech integrated the Disciple workaround generator into a larger system that can improve current approaches to air campaign planning by helping to automate analysis functions that previously could only be performed slowly and manually. In particular, it supports air campaign planning by a JFACC and his or her staff in two ways:

a) it facilitates the evaluation of targeting strategies aimed at disabling enemy infrastructure systems and
b) it provides information about enemy assets that should be preemptively targeted to impede their efforts to work around battle damage.

A Disciple-based solution to the course of action challenge problem ([24])

The Disciple approach has been further developed and scaled-up for application to the course of action (COA) challenge problem, in the second phase of the High Performance Knowledge Bases program. The developed Disciple-COA agent identifies strengths and weaknesses in a military course of action, based on the principles of war and tenets of army operations. This supports ground combat planning by the commander and staff in several ways:

- it identifies key combat tasks assigned to units;
- it analyzes the ability of units to accomplish their tasks;
- it evaluates the contributions of these tasks to the accomplishment of the mission.

With Disciple-COA, for the first time, the knowledge base of a Disciple agent was developed around an ontology created by another research group (Teknowledge and

Cycorp), demonstrating both the feasibility of knowledge reuse with the Disciple approach, and the generality of the Disciple rule learning and refinement methods. Moreover, the Disciple-COA agent was taught very rapidly by a knowledge engineer and a subject matter expert, and demonstrated higher performance than the other critiquers developed in the HPKB program. It also generated many correct COA critiques that were not anticipated by the evaluation experts.

Successful knowledge acquisition experiment ([22])

A customized version of Disciple-COA was used in a one-week knowledge acquisition experiment at the US Army Battle Command Battle Lab (BCBL) at Fort Leavenworth, Kansas. The main goal of the experiment was to demonstrate that it is possible for a military expert to teach Disciple how to critique a COA with respect to several principles of war. In this experiment, four military experts that did not have any prior knowledge engineering experience received around 16 hours of training in Artificial Intelligence and the use of Disciple-COA. They then succeeded in training Disciple to critique COAs with respect to the Principle of Offensive and the Principle of Security, starting with a KB containing the complete ontology of objects and features but no rules. During the training process that lasted around three hours, and without receiving significant assistance from knowledge engineers, each expert succeeded in extending the knowledge base of Disciple-COA with 28 tasks and 26 rules, following a model of the critiquing process that was provided to them at the beginning of the experiment. At the end of the experiment they completed a detailed questionnaire that revealed high scores for the perceived usefulness and usability of Disciple. For example, one of the experts stated: "The potential use of this tool by domain experts is only limited by their imagination—not their AI programming skills."

A domain modeling methodology based on task reduction ([7], [2])

A result of the performed research is the development of a simple and general domain modeling methodology that supports teaching-based intelligent agent development. The methodology is based on task reduction. It facilitates ontology specification, import and extension. It identifies the tasks to be represented in the agent's knowledge base. It guides the rule learning process, and also supports natural language generation of solutions and justifications by the agent. This methodology has been successfully applied in two different domains, workaround generation and course of action critiquing. Also, it was found to be natural and easy to use by the subject matter experts that participated in the knowledge acquisition experiment performed at the Battle Command Battle Lab, in 1999.

Knowledge representation for integrated problem solving and learning ([13], [22], [5])

In the Disciple approach, an agent's knowledge is represented using six types of knowledge elements: 1) Objects that represent either specific individuals or sets of individuals in the application domain; 2) Features and sets of features that are used to further describe objects, other features and tasks; 3) Tasks that represent anything that the

agent may be asked to accomplish; 4) Examples of task reductions; 5) Explanations of task reduction examples; and 6) IF-THEN plausible version space task reduction rules learned from examples.

The central element of the representation is the plausible version space task reduction rule. Instead of a single applicability condition, a rule specifies a plausible space of hypotheses for its condition. This plausible version space is represented by a plausible upper bound condition and by a plausible lower bound condition. During learning these two bounds will converge toward one another. In addition to the rule's condition that needs to hold in order for the rule to be applicable, the rule may have several "except-when" conditions that should not hold, in order for the rule to be applicable. The rule may also have "except-for" conditions (that specify instances that are negative exceptions of the rule) and "for" conditions (that specify positive exceptions). Much of the power of the Disciple approach comes from the original concept of plausible version space rule.

The ontology of objects, features and tasks serves as the generalization hierarchy for learning. An important aspect is that the ontology is itself evolving during knowledge acquisition and learning. This distinguishes Disciple from most of the other learning agents that make the less realistic assumption that the representation language for learning is completely defined before any learning can take place.

The objects, features and tasks are represented as frames, according to the knowledge model of the Open Knowledge Base Connectivity (OKBC) protocol. This facilitates the import of ontological knowledge from existing knowledge repositories that are OKBC-compliant.

A cooperative problem solving method ([22])

Based on the concept of a plausible version space rule, a cooperative problem solving method has been developed that facilitates agent training by a subject matter expert. Plausible version space rules are used in this cooperative problem solving process to generate task reductions with different degrees of plausibility, depending on which of its conditions are satisfied. If the plausible lower bound condition of a rule is satisfied, then the solution is very likely to be correct. If the plausible lower bound condition is not satisfied, but the plausible upper bound condition is satisfied, then the solution is considered only plausible. During the cooperative problem solving process the subject matter expert has to either accept a solution proposed by the agent (in which case the rule that generated this solution may be generalized), or to reject it (in which case the rule that generated the solution will be specialized). When the solution proposed by the agent is rejected by the expert, the expert has to provide the correct solution. From this expert solution the agent will learn a new plausible version space rule.

In addition to the cooperative problem solver, the Disciple shell also includes an autonomous problem solver.

Multistrategy rule learning and refining methods ([13], [3], [15])

A main research result is the development of two multistrategy learning methods, one for rule learning and another for rule refinement.

In the case of rule learning, the expert is teaching the agent how to solve a specific problem by providing a concrete example and by helping the agent to understand it. The agent uses learning from this example, from explanations and by analogy, to learn a general plausible version space rule that will allow it to solve similar problems. The expert can guide the agent to find the explanations of why the example is correct, by providing several types of hints. The explanations generated by the agent could contain complex relations between the objects from the agent's ontology, including numerical relations. The extensions of the current method with respect to its previous version include the use of more natural hints, more complex explanations, and methods to generalize these explanations.

In the case of rule refinement, the agent employs learning by analogy and experimentation, inductive learning from examples and learning from explanations. A significant extension of this method is the refinement of a rule with "except-when" conditions. An except-when condition will be generated when the rule was incorrectly applied in a certain situation and the agent found an explanation of why the solution indicated by the rule was wrong.

Exception-handling methods ([13], [11])

During knowledge base refinement the agent may encounter exceptions to a rule. One common cause of the exceptions is the incompleteness of the knowledge base that does not contain the terms to distinguish between the rule's examples and exceptions. This research has produced several exception handling methods that guide the expert to provide additional knowledge that will extend the representation space for learning such that, in the new space, the rules could be modified to remove the exceptions.

Intelligent educational agents ([19], [14])

The Disciple approach can naturally be used by a teacher to build certain types of educational agents. The educator can teach a Disciple agent which in turn can tutor students in the same way it was taught by the teacher. To demonstrate this claim, the Disciple approach has been used to build an educational agent that generates history tests for students. These tests provide intelligent feedback to the student in the form of hints, answer and explanations, and assist in the assessment of students' understanding and use of higher-order thinking skills.

An integration of machine learning and intelligent tutoring systems ([8])

The concept of a learning tutor has been defined as being an intelligent agent that learns from human tutors and then tutors human learners. The notion of a learning tutor provides a conceptual framework for integrating the fields of intelligent tutoring-learning environments and machine learning-based knowledge acquisition.

Mixed-initiative reasoning ([4], [6], [21])

Although the Disciple approach has been significantly developed, a subject matter expert still needs to receive a considerable amount of support from a knowledge engineer for several critical processes, such as domain modeling (which currently is an entirely manual process) or ontology import and development. Also, all the aspects of the Disciple approach need to be considerably simplified in order to achieve the vision of making agent development as easy as text processing or Internet browsing. Therefore, research has been started in developing the next generation of the Disciple theory, methodology and system, based on mixed-initiative reasoning that integrates complementary human and automated reasoning to take advantage of their respective knowledge, reasoning styles and computational strengths.

3. Cumulative list of written publications

[1] Boicu M., Wright K., Marcu D., Lee S.W., Bowman M. and Tecuci G., "The Disciple Integrated Shell and Methodology for Rapid Development of Knowledge-Based Agents," in *Proceedings of the Sixteenth National Conference on Artificial Intelligence* (AAAI-99), Intelligent Systems Demonstrations, July 18-22, Orlando, Florida, AAAI Press, Menlo Park, CA. 1999.

[2] Boicu M., Tecuci G., Bowman M., Marcu D., Lee S.W. and Wright K., "A Problem-Oriented Approach to Ontology Creation and Maintenance," in *Proceedings of the Sixteenth National Conference on Artificial Intelligence Workshop on Ontology Management*, July 18-19, Orlando, Florida, AAAI Press, Menlo Park, CA. 1999.

[3] Boicu M., Tecuci G., Marcu D., Bowman M., Shyr P., Ciucu F., and Levcovici C., "Disciple-COA: From Agent Programming to Agent Teaching," In *Proceedings of the Seventeenth International Conference on Machine Learning* (ICML), Stanford, California 2000, Morgan Kaufman.

[4] Boicu M. and Tecuci G., "Mixed-Initiative Reasoning for Integrated Domain Modeling, Learning and Problem Solving," in *Proceedings of the Seventeenth National Conference on Artificial Intelligence and the Twelfth Conference on Innovative Application of Artificial Intelligence*, Menlo Park, CA:AAAI Press, 2000.

[5] Boicu M., Tecuci G., Stanescu B., Panait L., Cascaval C., "Learnable Representation for Real World Planning," in *Proceedings of the AAAI-2000 Workshop on Representational Issues for Real-World Planning Systems*, Austin, Texas, 2000.

[6] Boicu M., Marcu D., Bowman M., and Tecuci G., "A Mixed-Initiative Approach to Teaching Agents to Do Things," In *Proceedings of the Symposium on Learning How to Do Things,* The 2000 AAAI Fall Symposium Series, North Falmouth, Massachusetts, November 3-5, 2000.

[7] Bowman M., Tecuci G., and Boicu M., "A Methodology for Modeling and Representing Expert Knowledge that Supports Teaching-Based Intelligent Agent Development," in *Proceedings of the Seventeenth National Conference on Artificial Intelligence and the Twelfth Conference on Innovative Application of Artificial Intelligence*, July 2000, Menlo Park, CA: AAAI Press.

[8] Hamburger H. and Tecuci G., "Toward a Unification of Human-Computer Learning and Tutoring," *Proceedings of the International Conference on Intelligent Tutoring Systems,* ITS'98, San Antonio, Texas, Springer Verlag, 1998.

[9] Hamburger H. and Tecuci G., "Architecture of a Pedagogical Agent for Human-Computer Learning and Tutoring," *ITS'98 Workshop 2 - Pedagogical Agents,* San Antonio, TX, 1998.

[10] Lee O., and Tecuci G., "MTLS: A Tool for Extending and Modifying Knowledge Bases," in *Proceedings of the Ninth IEEE International Conference on Tools with Artificial Intelligence* (ICTAI-97), Newport Beach, CA, November 5-7, 1997.

[11] Lee S.W. and Tecuci G., "Knowledge Base Revision through Exception-driven Discovery and Learning," in the *Proceedings of the Sixteenth National Conference on Artificial Intelligence* (AAAI-99), July 18-22, Orlando, Florida, AAAI Press, Menlo Park, CA. 1999.

[12] Rezazad H., Tecuci G., "Development of an Intelligent Agent for the Design of Local Area Networks," in *Proceedings of the 6th International Conference on Artificial Intelligence in Design*, Worcester Polytechnic Institute, Worcester, Massachusetts, USA, June 26-29, 2000.

[13] Tecuci G., *BUILDING INTELLIGENT AGENTS: An Apprenticeship Multistrategy Learning Theory, Methodology, Tool and Case Studies,* Academic Press, 1998.

[14] Tecuci G. and Keeling H., "Developing Intelligent Educational Agents with the Disciple Learning Agent Shell," in *Proceedings of the International Conference on Intelligent Tutoring Systems,* ITS'98, San Antonio, Texas, Springer Verlag, 1998 (best paper award).

[15] Tecuci G. and Keeling H., "Teaching an Agent to Test Students," in *Proceedings of the Fifteenth International Conference on Machine Learning,* Madison, Wisconsin, Morgan Kaufmann, 1998.

[16] Tecuci G., Wright K., Lee S.W., Boicu M., Bowman M. and Webster D., "A Learning Agent Shell and Methodology for Developing Intelligent Agents," *The AAAI-98 Workshop on Software Tools for Developing Agents,* Madison, Wisconsin, July 1988.

[17] Tecuci G., Keeling H., Dybala T., Wright K. and Webster D., "The Disciple Learning Agent Shell and a Disciple Test Generation Agent," *Exhibit at the International Conference on Intelligent Tutoring Systems,* ITS'98, San Antonio, Texas, 1998.

[18] Tecuci G. and Keeling H., "Efficient Development of Intelligent Test Generation Agents with the Disciple Learning Agent Shell," ITS'98 *Workshop 3 - Efficient ITS Development,* San Antonio, TX, 1998

[19] Tecuci G. and Keeling H., "An Experiment in Developing an Intelligent Educational Agent with the Disciple Learning Agent Shell," *International Journal of Artificial Intelligence in Education,* vol. 10, no.3-4, 1999.

[20] Tecuci G., Boicu M., Wright K., Lee S.W., Marcu D., and Bowman M., "An Integrated Shell and Methodology for Rapid Development of Knowledge-Based Agents," in *Proceedings of the Sixteenth National Conference on Artificial Intelligence* (AAAI-99), July 18-22, Orlando, Florida, AAAI Press, Menlo Park, CA. 1999.

[21] Tecuci G., Boicu M., Wright K. and Lee S.W., "Mixed-Initiative Development of Knowledge Bases," in *Proceedings of the Sixteenth National Conference on Artificial Intelligence Workshop on Mixed-Initiative Intelligence,* July 18-19, Orlando, Florida, AAAI Press, Menlo Park, CA. 1999.

[22] Tecuci G., Boicu M., Bowman M., Marcu D., Shyr P., and Cascaval C., "An Experiment in Agent Teaching by Subject Matter Experts," *International Journal of Human-Computer Studies*, 2000.

[23] Tecuci G., Boicu M., Wright K., Lee S.W., Marcu D. and Bowman M., "A Tutoring Based Approach to the Development of Intelligent Agents," in Teodorescu, H.N., Mlynek, D., Kandel, A. and Zimmermann, H.J. (editors). *Intelligent Systems and Interfaces*, Kluwer Academic Press. 2000.

[24] Tecuci G., Boicu M., Marcu D., Bowman M., Ciucu F., and Levcovici C., "Rapid Development of a High Performance Knowledge Base for Course of Action Critiquing," in *Proceedings of the Seventeenth National Conference on Artificial Intelligence and the Twelfth Conference on Innovative Application of Artificial Intelligence*, July 2000, Menlo Park, CA: AAAI Press.

[25] Tecuci G., Boicu M., Marcu D., "Learning Agents Teachable by Typical Computer Users," in *Proceedings of the AAAI-2000 Workshop on New Research Problems for Machine Learning*, Austin, Texas, 2000.

[26] Wright K., Boicu M., Lee S.W. and Tecuci G., "Building Agents from Shared Ontologies through Apprenticeship Multistrategy Learning," *The Fifteenth National Conference on Artificial Intelligence, AAAI-98,* Student Abstract Poster Program, Madison, Wisconsin, Morgan Kaufmann, 1998.

4. Presentations/demonstrations at AFOSR and DARPA meetings, invited talks

Tecuci G., HPKB meeting, Presentation of the GMU LALAB project, Boston, MA, April 1st, 1997.

Tecuci G., Wright K., Lee S.W., HPKB Kick-off meeting, Staunton, VA. Presentation and demo of the GMU LALAB project, June 4-6, 1997.

Tecuci G., HPKB Problem solving methods jump-start meeting, Presentation and demo of the GMU LALAB project, Los Angeles, CA, July 22-23, 1997.

Tecuci G., Wright K., Genoa knowledge acquisition meeting, Washington D.C., July 16, 1997.

Tecuci G., AFOSR PI Meeting, Presentation of the GMU LALAB research results, Rome Lab, September 9-11, 1997.

Tecuci G., HPKB-SAIC Integration Meeting, Presentation of the GMU LALAB project and current research results, San Diego, September 15-16, 1997.

Tecuci G., HPKB-TFS Integration meeting, Presentation of the GMU LALAB project and current research results, Palo Alto, September 17-18, 1997.

Tecuci G., Wright K., HPKB mid year PI meeting and workshop, San Diego, Presentation of the GMU project and demonstration of the Disciple prototype, December 3-5, 1997.

Tecuci G., Wright K., HPKB Subject matter expert meeting, Presentation of the GMU LALAB project and knowledge acquisition sessions with subject matter experts, Washington D.C., January 6-8, 1998.

Wright K., HPKB Subject matter expert meeting, Knowledge acquisition sessions with subject matter experts, Burlington, MA, February 1998.

Tecuci G., Wright K., HPKB Battlespace Ontology/Integration Meeting, Los Angeles, March 13, 1998.

Tecuci G., Invited talk: "Research on Building Intelligent Agents in the Learning Agents Laboratory," SPA, Alexandria, VA, January 12, 1998.

Tecuci G., Boicu M., Bowman M., Lee S.W., Wright K., HPKB Workshop, Presentation of the GMU year 1 results and demonstration of the GMU integrated system for workaround generation, Washington D.C., July 7-10.

Tecuci G., Wright K., HPKB COA Meeting, San Diego, September 22 - 25, 1998.

Tecuci G., Invited talk: "Building Intelligent Agents: An Apprenticeship Multistrategy Learning Approach," AI Seminar Series, Navy Center for Applied Research in Artificial Intelligence, October 19, 1998.

Tecuci G., Wright K., HPKB COA Meeting, Palo Alto, December 9-11, 1998.

Tecuci G., Boicu M., and Bowman M., HPKB Mid-year meeting, Presentation and Demonstration of GMU LALAB results, Austin, Texas, January 19-21, 1999.

Tecuci G., AFOSR PI Meeting, Presentation of the GMU LALAB research results, Colorado Springs, Colorado, February 2-4, 1999.

Tecuci G., Invited talk: "An Integrated Shell and Methodology for Rapid Development of Knowledge-Based Agents," Technical Speakers Series of the Information Systems and Technical Division of WC3, MITRE, February 18, 1999.

Tecuci G., HPKB Meeting, Presentation of GMU LALAB results, Los Angeles, April 7-9, 1998.

Bowman M., Presentation of the GMU LALAB project at Battle Command Battle Lab, Ft. Leavenworth, Kansas, April 13, 1999.

Tecuci G., Boicu M., Bowman M., Cascaval C., Ciucu F., Levcovici C., Marcu D., Panait L., Stanescu B., Final HPKB meeting, Presentation of the GMU LALAB research results, poster, and demonstration of the Disciple-COA system, Washington D.C., October 7-9, 1999

Tecuci G., Invited talk: "Teaching your Assistant - The Case of a Course of Action Critiquer," 1999 MITRE Knowledge and Agents Technical Exchange Meeting (KAT-99), November 15, 1999.

Tecuci G., Marcu D., AFOSR PI Meeting, Presentation of the GMU LALAB research results, ISI, Los Angeles, March 28-29, 2000.

5. Significant events and interactions

CYC course
GMU LALAB organized a two day course on the CYC system for the East Coast participants of the HPKB program, Fairfax, VA, October 28 – 29, 1997.

Subject matter expert meeting
GMU LALAB organized a two day meeting with a subject matter expert on April 23-24, 1998, for the HPKB teams addressing the workaround challenge problem.

Best results at 1998 HPKB annual evaluation
GMU LALAB demonstrated the highest rate of knowledge acquisition and the best quality of the generated solutions during the HPKB workaround challenge problem evaluation (June 17 – July 1, 1998).

Best paper award at ITS'98
Tecuci G. and Keeling H., "Developing Intelligent Educational Agents with the Disciple Learning Agent Shell," received "The Best Paper Award" at the International Conference on Intelligent Tutoring Systems (ITS-98), San Antonio, Texas.

Contribution to EFX'98
Disciple-workaround was demonstrated by Alphatech at EFX'98, as part of a larger system for the evaluation of targeting strategies.

Book on the Disciple approach published by Academic Press
Academic Press published the book "*BUILDING INTELLIGENT AGENTS: An Apprenticeship Multistrategy Learning Theory, Methodology, Tool and Case Studies,*" Tecuci G., 1998, which represented a comprehensive description of the Disciple approach.

Participation at the development of the COA integrated system
GMU LALAB collaborated with Teknowledge, Cycorp, Alphatech, Northwestern Univ, SAIC and Univ. of Edinburgh to develop an end-to-end integrated system for COA critiquing.

Best results at 1999 HPKB annual evaluation
GMU LALAB obtained the best results at the HPKB annual evaluation of the COA critiquers (July 8 - July 16, 1999).

Successful knowledge acquisition experiment at BCBL
GMU LALAB conducted a one week (August 23 – August 27, 1999) knowledge acquisition experiment at the US Army Battle Command Battle Lab, in Ft. Leavenworth, KS. In the experiment, four military experts with no prior knowledge engineering experience received very limited training in the teaching of Disciple-COA and then each succeeded to significantly extend its KB, receiving only very limited support from a knowledge engineer.

6. Personnel associated with the research effort

Faculty:	Gheorghe Tecuci
Students:	Mihai Boicu, Mike Bowman, Cristina Cascaval, Florin Ciucu, Harry Keeling, Seok-Won Lee, Cristian Levcovici, Dorin Marcu, Liviu Panait, Ping Shyr, Bogdan Stanescu, Kathryn Wright